IN THE CLAIMS:

Please amend the claims as follows:

1. (Currently Amended) A method of context-sensitive word validity checking in a programming environment, comprising:

receiving user input information at an input location in the programming environment;

determining a context of the input location, wherein determining the context comprises determining a cursor location and determining a scope for the cursor location, wherein the scope is defined by a start position and an end position in the programming environment;

determining a plurality of relevant terms selected according to the context; and

determining a validity of the user input information against the plurality of relevant terms.

2. (Original) The method of claim 1, wherein the plurality of relevant terms are determined according to at least one of valid syntax and spelling at the input location.

3. (Original) The method of claim 1, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a program method; and

if so, selecting a predefined dictionary containing executable method code which is syntactically correct for program methods.

4. (Original) The method of claim 1, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a comment; and

if so, selecting a predefined spoken language dictionary comprising non-executable terms.

5. (Original) The method of claim 4, wherein determining the plurality of relevant terms further comprises:

determining whether the input location is in a program method;

if so, selecting a first predefined dictionary containing executable method code which is syntactically correct for program methods; and

If the location is neither in a comment nor a program method, selecting a second predefined dictionary containing executable non-method code which is syntactically correct for locations outside program methods.

6. (Original) The method of claim 1, further comprising visually indicating invalid usages of the user input information.

7. (Original) The method of claim 1, wherein determining the plurality of relevant terms comprises:

identifying the relevant terms according to the context; and

dynamically populating a list with the relevant terms.

8. (Currently Amended) The method of claim 1, wherein ~~determining the context comprises determining a cursor location~~ the scope is determined based on a scope table containing index data of scope start positions and scope end positions.

9. (Currently Amended) The method of claim 8, ~~wherein determining the context further comprises determining a scope for the cursor location~~ further comprising updating the index data in the scope table reflecting user input into the programming environment.

10. (Currently Amended) The method of claim 1 [[9]], wherein the plurality of relevant terms change in response to moving the cursor location to another scope.

11. (Original) The method of claim 10, wherein the plurality of relevant terms remain unchanged with a change in the cursor location within the scope.

12. (Currently Amended) A computer, comprising:

a memory containing a development tool, a word validity checker and at least one variable dictionary and at least one static dictionary, wherein the at least one variable dictionary is configured to contain user-defined terms specific to the programming environment and the at least one static dictionary contains terms persistent between programming environments;

a processor which when configured with the development tool processes user input in a programming environment and when configured with the word validity checker is configured to perform an operation comprising:

in response to receiving user input information at an input location in the programming environment, determining a plurality of relevant terms selected according to the input location wherein at least a portion of the relevant terms are selected from the at least one static dictionary and a portion of the relevant terms are stored to the at least one variable dictionary, wherein determining the plurality of relevant terms comprises determining a scope for the input location, wherein the scope is defined by a start position and an end position in the programming environment; and

determining a validity of the user input information against the plurality of relevant terms.

13. (Original) The computer of claim 12, further comprising an output device and wherein the processor is configured to cause a visual indication of invalid usages of the user input information on the output device.

14. (Original) The computer of claim 12, wherein the processor is configured to determine the plurality of relevant terms according to at least one of valid syntax and spelling at the input location

15. (Original) The computer of claim 12, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a program method; and

if so, selecting one dictionary from the at least one static dictionary containing executable method code which is syntactically correct for program methods.

16. (Original) The computer of claim 12, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a comment; and

if so, selecting a spoken language dictionary from the at least one static dictionary comprising non-executable terms.

17. (Original) The computer of claim 16, wherein determining the plurality of relevant terms further comprises:

determining whether the input location is in a program method;

if so, selecting a first dictionary from the at least one static dictionary containing method executable code which is syntactically correct for program methods; and

if the location is neither in a comment nor a program method, selecting a second dictionary from the at least one static dictionary containing non-method executable code which is syntactically correct for locations outside program methods.

18. (Currently Amended) The computer of claim 12, wherein determining the ~~plurality of relevant terms comprises determining a~~ scope for the input location is based on a scope table containing index data of scope start positions and scope end positions.

19. (Currently Amended) The computer of claim [[18]] 12, wherein the processor is configured to identify another scope in response to moving a cursor location to the another scope.

20. (Currently Amended) A computer readable medium containing a word validity check program which when executed, performs an operation of context-sensitive word validity checking in a programming environment, the operation comprising:

in response to receiving user input information at an input location in the programming environment, determining a context of the input location, wherein

determining the context comprises determining a cursor location and determining a scope for the cursor location, wherein the scope is defined by a start position and an end position in the programming environment;

determining a plurality of relevant terms selected according to the context; and

determining a validity of the user input information against the plurality of relevant terms.

21. (Original) The computer readable medium of claim 20, wherein the operation further comprises outputting a visual indication of any invalid user input information.

22. (Original) The computer readable medium of claim 20, wherein the plurality of relevant terms are determined according to at least one of valid syntax and spelling relative to the input location.

23. (Original) The computer readable medium of claim 20, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a program method; and

if so, selecting a predefined dictionary containing executable method code which is syntactically correct for program methods.

24. (Original) The computer readable medium of claim 20, wherein determining the plurality of relevant terms comprises:

determining whether the input location is in a comment; and

if so, selecting a predefined spoken language dictionary comprising non-executable terms.

25. (Original) The computer readable medium of claim 24, wherein determining the plurality of relevant terms further comprises:

determining whether the input location is in a program method;

if so, selecting a first predefined dictionary containing executable method code which is syntactically correct for program methods; and

if the location is neither in a comment nor a program method, selecting a second predefined dictionary containing executable non-method code which is syntactically correct for locations outside program methods.

26. (Currently Amended) The computer readable medium of claim 20, wherein ~~determining the context comprises determining a cursor location~~ the scope is determined based on a scope table containing index data of scope start positions and scope end positions.

27. (Currently Amended) The computer readable medium of claim 20, ~~wherein determining the context comprises:~~

~~determining a cursor location; and~~

~~determining a scope for the cursor location.~~

further comprising updating the index data in the scope table reflecting user input into the programming environment.

28. (Currently Amended) The computer readable medium of claim [[27]] 20, wherein the plurality of relevant terms change in response to moving the cursor location to another scope.

29. (Currently Amended) The computer readable medium of claim [[27]] 28, wherein the plurality of relevant terms remain unchanged with a change in the cursor location within the scope.

30. (Currently Amended) A computer readable medium containing a word validity check program which when executed, performs an operation of context-sensitive word validity checking in a programming environment, the operation comprising:

in response to receiving user input information at an input location in the programming environment, determining a context of the input location, wherein determining the context comprises determining a scope according to a cursor location,

wherein the scope is defined by a start position and an end position in the programming environment;

determining a validity of the user input information relative to the context; and

outputting a visual indication of any invalid user input information.

31. (Original) The computer readable medium of claim 30, wherein determining the validity comprises determining at least one of valid syntax and spelling relative to the input location.

32. (Original) The computer readable medium of claim 30, wherein determining the context comprises determining whether the input location is in a program method and, if so, wherein the validity comprises determining whether the user input information is syntactically correct method code.

33. (Original) The computer readable medium of claim 30, wherein determining the context comprises determining whether the input location is in a comment and, if so, wherein determining the validity comprises determining whether the user input information is spelled correctly according to a spoken language dictionary and a program language keyword dictionary.

34. (Currently Amended) The computer readable medium of claim 30, wherein ~~determining the context comprises determining a scope according to a cursor location~~ the scope is determined based on a scope table containing index data of scope start positions and scope end positions.